

(19) World Intellectual Property Organization  
International Bureau



(43) International Publication Date  
7 August 2008 (07.08.2008)

PCT

(10) International Publication Number  
**WO 2008/094728 A1**

(51) International Patent Classification:  
**G06K 19/07** (2006.01)

[US/US]; 3225 Cayman Place, Boulder, Colorado, US  
80301 (US).

(21) International Application Number:  
PCT/US2008/050630

(74) Agent: **WARD, John P.**; Greenberg Traurig, LLP, 2450  
Colorado Avenue, Suite 400 East, Santa Monica, California  
90404 (US).

(22) International Filing Date: 9 January 2008 (09.01.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:  
11/700,525 30 January 2007 (30.01.2007) US

(71) Applicant (for all designated States except US):  
**KEYSTONE TECHNOLOGY SOLUTIONS, LLC.**  
[US/US]; 8000 South Federal Way, Boise, Idaho 83716  
(US).

(81) Designated States (unless otherwise indicated, for every  
kind of national protection available): AE, AG, AL, AM,  
AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA,  
CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE,  
EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID,  
IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC,  
LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN,  
MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH,  
PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV,  
SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN,  
ZA, ZM, ZW.

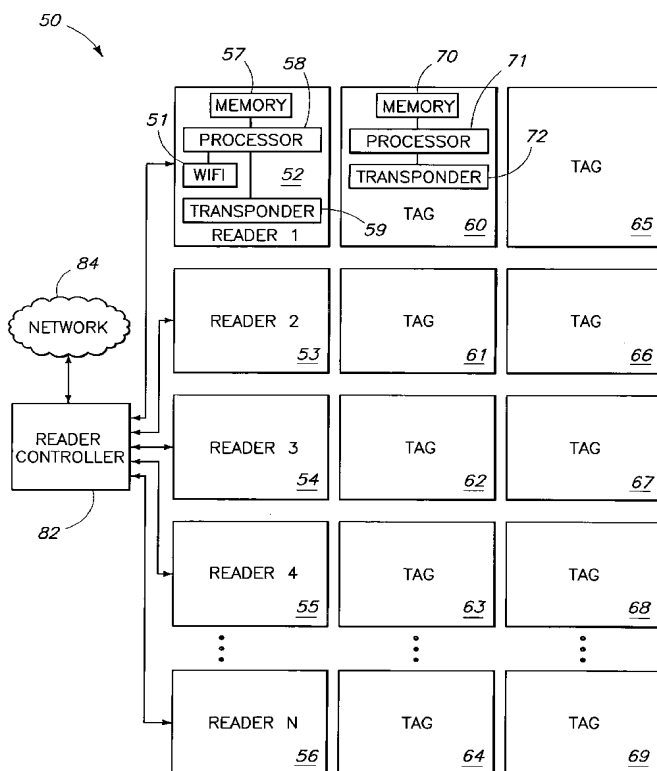
(72) Inventor; and

(75) Inventor/Applicant (for US only): **TUTTLE, John R.**

(84) Designated States (unless otherwise indicated, for every  
kind of regional protection available): ARIPO (BW, GH,  
GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM,

[Continued on next page]

(54) Title: SYSTEMS AND METHODS FOR RFID TAG ARBITRATION



(57) Abstract: An RFID (radio frequency identification device) tag comprising a processor; a memory configured to store an identification number that distinguishes the tag from other tags; a transponder coupled to the memory and the processor, the tag being configured to arbitrate by selecting a random number, in response to an inventory query from a reader, and to respond to the reader depending on the random number selected, the tag being further configured for multiple concurrent inventory session arbitrations with multiple readers by separately storing random numbers for respective inventory session arbitrations. RFID readers, systems, and methods are also provided.

WO 2008/094728 A1



ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM),  
European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI,  
FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL,  
NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG,  
CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

**Published:**

- *with international search report*
- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

## SYSTEMS AND METHODS FOR RFID TAG ARBITRATION

### TECHNICAL FIELD

**[0001]** The technical field relates to radio frequency identification. More particularly, various embodiments of the invention relate to methods and apparatus for inventorying radio frequency identification devices.

### BACKGROUND

**[0002]** Radio frequency identification devices (RFIDs) are known in the art. Such devices are typically used for inventory tracking. As large numbers of objects are moved in inventory, product manufacturing, and merchandising operations, there is a continuous challenge to accurately monitor the location and flow of objects. Additionally, there is a continuing goal to determine the location of objects in an inexpensive and streamlined manner. One way to track objects is by affixing RFID tags to objects or groups of objects, and interrogating the RFID tags with an interrogator or reader to determine which objects are present in any particular location. RFID tags may be provided with unique identification numbers or codes in order to allow a reader to distinguish between multiple different tags.

**[0003]** Some RFID tags use the electromagnetic field from an interrogator for power. Typically, these devices are passive (have no power supply), which results in a small and portable package.

**[0004]** Another type of RFID tag is an active RFID tag, which includes its own source of power, such as a battery.

**[0005]** If an interrogator or reader has prior knowledge of the identification number of a device, the reader can specify that a response is requested only from the device with that identification number. Sometimes, such information is not available. For example, there are occasions where a reader is attempting to determine which of multiple devices are within communication range. When the reader sends a message to a transponder device requesting a reply, there is a possibility that multiple transponder devices will attempt to respond

simultaneously, causing a collision, and thus an erroneous message to be received by the reader. For example, if the interrogator sends out a command requesting that all devices within a communications range identify themselves, and receives a large number of simultaneous replies, the interrogator may not be able to interpret any of these replies. Tag responses can interfere with each other and the reader sees collisions or incomprehensible noise. Therefore, arbitration or singulation schemes are employed to permit communications that are free of collisions. The term singulation refers to identifying a specific individual tag in a multiple tag environment.

**[0006]** In some arbitration or singulation schemes, described in commonly assigned U.S. Patent Nos. 5,627,544; 5,583,850; 5,500,650; and 5,365,551, all to Snodgrass et al. and the disclosures of all of which are incorporated herein by reference, a reader sends a command causing each device of a potentially large number of responding devices to select a random number from a known range and use it as that device's arbitration number. By transmitting requests for identification to various subsets of the full range of arbitration numbers, and checking for an error-free response, the interrogator quickly determines the arbitration number of every responder station capable of communicating at the same time. Thereafter, the interrogator is able to conduct subsequent uninterrupted communication with devices, one at a time, by addressing only one device. Various arbitration or singulation schemes are discussed in commonly assigned U.S. Patent Nos. 6,275,476 to Wood, Jr.; 6,118,789 to Wood, Jr.; 6,072,801 to Wood, Jr. et al.; and 6,061,344 to Wood, Jr., the disclosures of all of which are incorporated herein by reference.

**[0007]** It is possible to have multiple readers operating in the same location. Problems can arise when multiple readers try to conduct an inventory at the same time.

**[0008]** EPCglobal is a standard setting organization that is developing standards for electronic product codes to support the use of RFID technology. One of their standards, called Class 1, Generation 2 (also known as "Gen 2") applies to passive RFID systems, and is described on their websites at [www.epcglobalus.org](http://www.epcglobalus.org) or [www.epcglobalinc.org](http://www.epcglobalinc.org). These standards evolve over time, and for a particular standard, such as Gen 2, there are minor variations between versions. The present version of the Class 1, Generation 2 standard is version 1.0.9.

[0009] The various protocols are designed to arbitrate the collisions, and EPCglobal's Gen 2 provides a specific methodology for arbitrating the collisions. In some of the above described patents, a binary-tree approach is used, in which some unique identifier is assigned to each tag, such as the electronic product code, and the reader goes down the tree of possible numbers, until it is confident that it is talking to just a single tag. At that point the tag is read, and put to sleep. The reader then goes back up the tree and tries to singulate another tag. This repeats until there are no more tags left responding.

[0010] The Gen 2 standard takes a different approach to generating an inventory. The approach used by the Gen 2 standard, has inventory rounds of  $2^Q$  slots. Although the word "slot" is used in the specification, it is not really a timeslot. It is more of a counter. The process is illustrated in Fig. 1. At the beginning of each round, each tag sets its respective slot counter to a random number, from 0 to  $2^Q-1$ . The reader sends a command that starts an inventory process. Any tag whose slot counter is 0 sends a reply; all other tags decrease their slot counter by 1. This process is repeated for all  $2^Q$  slots.  $Q$  is the number of bits, or width of the slot counter. By changing  $Q$ , the reader can optimally adjust the number of slots per round to adapt to the number of tags expected to be present. For example, if it is known that there are no more than six tags in a field, there is no need to use a slot counter that is 16 bits wide when 3 bits are sufficient. Thus  $Q$ , of the Gen 2 standard, is effectively a mask on the slot counter.  $Q$  is similar to the width of the Arbitration Mask described in the above-incorporated Snodgrass and Wood, Jr. patents.

[0011] Thus,  $Q$  is a parameter that is used to regulate the probability of tag response. During an inventory round, the tags respond only when their slot counter equals zero. The slot counter contents are derived in a pseudorandom manner based on the value of the parameter  $Q$ . In the Gen 2 standard,  $Q$  is, for example, an integer between 0 and 15, and the number of slots is between  $2^0$  and  $2^{15}$ .

[0012] The Gen 2 standard further allows the user to include in an inventory round only tags that meet certain selection criteria. Appropriate combinations of Select commands can be used to implement Boolean criteria within a tag population.

**[0013]** The Gen 2 standard has also implemented a method called “Sessions” to attempt to solve the problem of two, three or four readers reading the same population of tags in the same time period; i.e., an inventory processes overlap. According to the specification, a reader shall support and tags shall provide four sessions, and tags shall participate in one and only one session during an inventory round. Two or more interrogators can use sessions to independently inventory a common tag population. Tags associate a separate and independent “inventoried” flag to each of the multiple readers. After singulating a tag, an interrogator may issue a command that causes the tag to invert its inventoried flag for that session. However, all four sessions use the same random number generator and same slot counter. When a tag’s arbitration procedure gets interrupted (when one sessions stops and another session starts), the tag has to start over in an arbitration with a new reader and has to generate a new slot counter and handle (RN16) value. When the slot counter gets to zero, the tag replies and may have been inventoried by the interrupting reader. But when the first reader tries to resume its inventorying, the tag has lost its original handle and the reader cannot find the tag. The first reader has to begin its inventory process all over again.

**[0014]** Relevant portions of the Gen 2 specification will now be substantially repeated, to better enable one of ordinary skill in the art to understand the Gen 2 arbitration procedure. According to the Gen 2 specification, the inventory commands that a Gen2 reader or interrogator use include a “Query” command, a “QueryAdjust” command (described below), a “QueryRep” command (described below), an “ACK” (acknowledge) command, and a “NAK” (no acknowledgement or not acknowledged) command or transmission. The Query command initiates an inventory round and decides which tags participate an inventory round. A round is a period between successive Query commands.

**[0015]** Also according to the Gen 2 specification, the Query command contains the slot-count parameter  $Q$ . Upon receiving a Query command, participating tags pick a random number in the range of 0 to  $2^Q-1$ , inclusive, and load this value into their slot counter. Tags that pick a zero transition to the reply state and reply immediately. Tags that pick a nonzero value transition to an “arbitrate” state (see Fig. 3) and await a QueryAdjust (described below) or a QueryRep command. Assuming a single tag replies, the following occurs:  
a) the tag backscatters an RN16 (16 bit random or pseudo random number) as it enters a

reply state, b) the reader acknowledges the tag with an ACK COMMAND containing the same RN16, c) the acknowledged tag transitions to an “acknowledged” state, and backscatters its PC (protocol control data), EPC (electronic product code data that identifies the object to which the tag is or will be attached and distinguishes the tag from other tags) and CRC-16 (cyclic redundancy check), and d) the interrogator issues a QueryAdjust or QueryRep command (not shown), causing the identified tag to invert an “inventoried” flag (i.e., from A to B or B to A) and to transition to a “ready” state, and potentially causing another tag to initiate a query-response dialog with the interrogator, starting in step (a), above. The tag can be considered to be singulated after it is acknowledged.

**[0016]** If the tag fails to receive the ACK acknowledgement in step (b) within a specified time, or receives the ACK with an erroneous RN16, it returns to the arbitrate state. If multiple tags reply in step (a) above, but the reader is able to resolve an RN16 from one of the tags, the reader can acknowledge the resolved tag. Unresolved tags receive erroneous RN16s from the reader and return to the arbitrate state without backscattering their PC, EPC, and CRC-16.

**[0017]** If the reader sends a valid acknowledgement (i.e., an ACK containing the correct RN16) to the tag in the acknowledged state, the tag re-backscatters its PC, EPC, and CRC-16.

**[0018]** At any point the reader may issue a NAK. In response to receiving the NAK, all tags in the inventory round return to the arbitrate state without changing their inventoried flag.

**[0019]** After issuing a Query command to initiate an inventory round, the reader typically issues one or more QueryAdjust or QueryRep commands. The QueryAdjust repeats a previous Query and may increment or decrement Q, but does not introduce new tags into the round. QueryRep repeats a previous Query without changing any parameters and without introducing new tags into the round. An inventory round can contain multiple QueryAdjust or QueryRep commands. At some point the reader will issue a new Query, thereby starting a new inventory round.

**[0020]** Tags in the arbitrate or reply states that receive a QueryAdjust first adjust Q (increment, decrement, or leave unchanged), then pick a random value in the range of 0 to

$2^Q-1$ , inclusive, and load this random value into their slot counter. Tags that pick zero transition to the reply state and reply immediately. Tags that pick a nonzero value transition to the arbitrate state and await a QueryAdjust or a QueryRep command.

**[0021]** Tags in the arbitrate state decrement their slot counter every time they receive a QueryRep command, transitioning to the reply state and backscattering an RN16 when their slot counter reaches 0000 (hexadecimal). Tags whose slot counter reached 0000, who replied, and who were not acknowledged (including tags that responded to the original Query and were not acknowledged) return to the arbitrate state with a slot value of 0000 and decrement this slot value from 0000 to 7FFF (hexadecimal) at the next QueryRep, thereby effectively preventing subsequent replies until the tag loads a new random value into its slot counter. Tags reply at least once in  $2^Q-1$  QueryRep commands.

**[0022]** Although tag inventory is based on a random protocol, the Q-parameter affords network control by allowing a reader to regulate the probability of tag responses. Q is an integer in the range of (0,15) thus, the associated tag-response probabilities range from  $2^0=1$  to  $2^{-15}=0.000031$ .

**[0023]** To illustrate an inventory operation, the Gen 2 specification provides the following specific example. Assume a population of sixty-four powered tags in the ready state. A reader first issues a Select command to select a subpopulation of tags. Assume that sixteen tags match the selection criteria. Further, assume that twelve of the sixteen selected tags each have their inventoried flag set to A in session S0. The reader issues a Query specifying (SL, Q=4, S0, A). Each of the 12 tags having their inventoried flag set to A picks a random number in the range of (0,15) and loads the value into its slot counter. Tags that pick a zero respond immediately. The Query has three possible outcomes:

- a) No tags reply. The reader may issue another Query, or it may issue a QueryAdjust or QueryRep.
- b) One tag replies. See Fig. 2. The tag transitions to the reply state and backscatters 20 an RN16. The reader acknowledges the tag by sending 22 an ACK. If the tag receives the ACK with a correct RN16, it backscatters 24 its PC, EPC, and CRC-16 and transitions to the acknowledged state. If the tag receives the ACK with an incorrect RN16, it transitions to arbitrate. Assuming a successful ACK, the reader may either access the acknowledged



tag or issue 26 a QueryAdjust or QueryRep to invert the tag's inventoried flag from A to B and send the tag to the ready state. A Query with a matching prior-round session parameter will also invert the inventoried flag from A to B.

c) Multiple tags reply. The reader observes a backscattered waveform comprising multiple RN16s. It may try to resolve the collision and issue an ACK; not resolve the collision and issue a QueryAdjust, QueryRep, or NAK; or quickly identify the collision and issue a QueryAdjust or QueryRep before the collided tags have finished backscattering. In the latter case, the collided tags, not observing a valid reply within the specified time, return to the arbitrate state and await the next Query of QueryAdjust command. The Gen 2 arbitration process is also described in connection with a tag state diagram included in the Gen 2 specification, a relevant portion of which is reproduced as Fig. 3. Arbitrate state 30, reply state 32, and acknowledged state 34 correspond to such states as discussed above. Fig. 3 also shows that the term "handle," as used in the Gen 2 specification, corresponds to an RN16 that a reader uses as a nickname for a tag after a tag has a zero slot value and is acknowledged. Additional details, if desired, can be seen in the Gen 2 specification.

**[0024]** Instead of asking for a tag with a specific masked random number as described in the above incorporated Snodgrass and Wood, Jr. patents (and the reader subsequently going up or down through all numbers in the masked random number space, e.g., using the logic trees of the Wood, Jr. patents), the slot counters of tags based on the Gen 2 specification count down toward 0 in response to receiving each QueryRep command. When a tag slot counter reaches 0, the tag generates a new random number and returns it to the reader, and this is used as a temporary tag number or "handle" by the reader. The Gen 2 handle is substantially similar to the Random Number of the Snodgrass and Wood, Jr. patents. The Gen 2 process is substantially similar to the Snodgrass and Wood Jr. processes except that the Gen 2 tags make the countdown steps and issue new and large random numbers. This requires significantly more complexity in the tags, where real estate is expensive, instead of in the readers.

**[0025]** A conventional RFID tag 36, shown in Fig. 4, has one random number generator 38, one location 40 for storing random numbers, and one mask or Q value storage location 42. There is a problem when a second reader 44 initiates an arbitration process concurrent

with the arbitration process of a first reader 46. The second reader 44, while it may have a different session number, must command tags to each generate a new random number and mask. The new random numbers and new masks may differ in value from those generated for the first reader arbitration. Since each tag 36 stores only one random number and one mask or Q value, the first random number and mask are lost to the tags and unavailable to the first reader 46 for use upon continuing (resuming) after a second reader's arbitration process. The first reader 46 must send commands for its tag population to generate new random numbers and perhaps different masks (Q values), both of which can lead to prolonged inventory times. The problem is exacerbated if an additional reader 48 is introduced. In some cases, the inventory processes may never finish. In other words, the EPCglobal system does not truly provide multiple concurrent inventorying, even for two sessions, let alone four.

**[0026]** More particularly, using conventional tags in multiple concurrent arbitrations will result in extra sortings of tags. This is because, when a second reader starts its sort, the second reader's command will cause the tags to generate new random numbers and Q values that are, at that point, lost to the first reader's sorting process. This causes the first reader to ask for a new random number that is masked to the first Q value. The first (interrupted) sorting addressing sequence is lost and must be restarted upon resuming. In the case of multiple interruptions and restarts, the arbitration process of any reader may in some cases be extended indefinitely and not converge to a conclusion.

#### BRIEF DESCRIPTION OF THE DRAWINGS

**[0027]** Fig. 1 is a timing diagram illustrating an inventory operation and slot counter operation in accordance with the EPCglobal method.

**[0028]** Fig. 2 is a communication sequence diagram illustrating data communications between a tag and a reader if a single tag picks a zero value for its slot counter.

**[0029]** Fig. 3 is a state diagram illustrating states that a tag may go through as part of an arbitration process.

**[0030]** Fig. 4 is a block diagram illustrating problems with prior art designs.

[0031] Fig. 5 is a block diagram illustrating a system in accordance with various embodiments of the invention.

[0032] Fig. 6 is a block diagram illustrating data stored in a tag of the system of Fig. 5, for multiple inventory sessions, in some embodiments.

[0033] Fig. 7 is a block diagram illustrating data stored in a reader of the system of Fig. 5, in some embodiments.

#### DETAILED DESCRIPTION OF THE ILLUSTRATED EMBODIMENTS

[0034] Some embodiments of the invention relate to increasing the efficiency of multiple concurrent arbitration of an RFID tag population. Some embodiments of the invention relate to improving re-entry into an interrupted inventory session.

[0035] Some embodiments provide a method to enhance RFID tag arbitration which utilizes a random number generator (RNG), RNG register, RNG mask (Q value), and mask register, for each of a multiplicity of concurrent tag arbitration processes; thereby increasing the efficiency and effectiveness of each individual arbitration process and overall arbitration processes for inventorying multiple concurrent tag populations, while preventing the case of infinite loop non-convergence of the processes.

[0036] Fig. 5 shows a system 50 in accordance with some embodiments of the invention. The system 50 includes one or more tags and one or more readers. More particularly, in the embodiment shown in Fig. 5, the system 50 includes readers 52, 53, 54, 55, and 56, and tags 60, 61, 62, 63, 64, 65, 66, 67, 68, and 69. Additional or fewer readers can be included. Additional or fewer tags could be included. Further, there is no necessary relationship between the number of tags and the number of readers.

[0037] In some embodiments, the system 50 includes a reader controller 82 controlling the readers 52-56. The reader controller 82 permits control of the various readers as well as communication of data between the various readers. The reader controller 82 can further be used for reader multiplexing as described in applicant's commonly assigned patent application titled "RFID Communications Systems and Methods, And RFID Readers and Systems" by John Tuttle (attorney docket no. MI22-3366). In some embodiments, the reader controller 82 may be coupled to a network 84. This could be useful for control from

a remote location or via the Internet, for example. In some embodiments, the readers 52-56 may communicate with each other directly (or via an intermediate reader), instead of through a reader controller 82, whether or not the reader controller 82 is included. For example, the readers may be connected by hard wire or wirelessly (e.g., by Wi-Fi or other wireless communication). In some embodiments, the reader other embodiments, the reader controller 82 is omitted.

**[0038]** Respective tags 60-69 include memory 70, a processor 71, and a transponder or transceiver 72 (e.g., including a backscatter modulator). In some embodiments, one or more of the tags include all of the hardware required by the Gen 2 specification. In other embodiments, one or more of the tags include some subset of the hardware required by the Gen 2 specification.

**[0039]** Respective readers 52-56 include memory 57, a processor 58, and a transponder or transceiver 59 for communications with tags 60-69. The readers may also include a Wi-Fi or other wireless transceiver 51 for wireless communications with other readers, in addition to the transponder 59.

**[0040]** Fig. 6 shows data stored in respective tags 60-69, in some embodiments. One tag 60 is illustrated in Fig. 6, for simplicity. The data can be stored in the memory 70, for example (see Fig. 5).

**[0041]** Some embodiments provide, in respective tags 60-69, separate memory locations for random number or handle values 86, 88, 90, 92, and 94 for each arbitration or singulation process 96, 98, 100, 102, and 104. These correspond, for example, to the RN16s of Gen 2, though they may have more or less than 16 bits. The random numbers or handles may be stored in any appropriate register or memory location on the tags. The random number or handle values may change and are affected by commands from a reader, as specified in the Gen 2 specifications for example. However, the last used (most recent) random number or handle values for each inventory session is stored in each tag

**[0042]** Some embodiments provide, in respective tags 60-69, separate memory locations for Q values or arbitration mask values 106, 108, 110, 112, and 114 for each arbitration or singulation process 96, 98, 100, 102, and 104. The Q value indicates the width, in number of bits, of the slot counter, in some embodiments. The Q value or mask may be stored in

any appropriate register or memory location on the tags. The Q values may change and are affected by commands from a reader, as specified in the Gen 2 specifications for example. However the last used (most recent) Q value or arbitration mask for each inventory session 96, 98, 100, 102, and 104 is stored in each tag.

**[0043]** Some embodiments provide, in respective tags 60-69, separate slot counters 116, 118, 120, 122, and 124 for respective arbitrations. Respective tags are configured to reply to a reader if the value in a slot counter associated with a current or active arbitration is zero. The slot values are selected at random and are decremented in response to QueryAdjust or QueryRep commands, as set forth in the Gen 2 specification.

**[0044]** In some embodiments, respective tags 60-69 also store information, in memory locations 126, 128, 130, 132, and 134, identifying the session or reader associated with the stored last slot counter, last Q value, and last random number value or handle for a session.

**[0045]** The inventory sessions 96, 98, 100, 102, and 104 would typically represent different readers 52-56. However, it is possible for a second reader to complete an interrupted inventory session of a first reader if data from a first inventory session is passed on to the second reader either directly from the first reader or from the reader controller 82.

**[0046]** Fig. 7 shows data stored in respective readers 52-56, in some embodiments. One reader 52 is illustrated in Fig. 7, for simplicity. The data can be stored in the memory 57, for example (see Fig. 5). The data shown in Fig. 7 and described below can alternatively or additionally be stored in the reader controller 82. Further, the data for one reader's arbitration can also be passed to and stored in another reader.

**[0047]** In the embodiment shown in Fig. 7, the respective readers 52-56 store information, in memory locations 140, 142, 144, 146, 148, and 150, for handles for previously singulated tags, the handle being a nickname for a tag and being different from the tag's electronic product code. The handle corresponds, for example, to the RN16 of the Gen 2 specification, though more or less than 16 bits are possible. If an arbitration is interrupted, the reader is able to resume the arbitration without a need to cause singulated tags to generate new handles, even if another reader performs another arbitration session and causes tags to generate handles. This is because handles for previous arbitrations are stored and not lost due to a new arbitration process.

**[0048]** In some embodiments, the readers 52-56 also store information, in memory locations 152, 154, 156, 158, 160, and 162 for last used Q values or arbitration masks for respective arbitrations. In some embodiments, no Q value is used or stored in the reader or tags. For example, a Q value of all 1 bits could be assumed, resulting in no masking taking place.

**[0049]** Some embodiments provide, in respective readers 52-56, memory locations 164, 166, 168, 170, 172, and 174 for slot counter values for previously singulated tags. The slot counter values are used to affect the order in which tags attempt to communicate with a reader in an arbitration. When slot counter values reach zero, a tag responds to a reader in an active arbitration, as described above.

**[0050]** Some embodiments provide, in respective readers 52-56, memory locations 176, 178, 180, 182, 184, and 186 for session number identifying an (e.g., interrupted) arbitration. Q values, handles, and slot counter values for tags that were singulated in an arbitration session are associated with the session number for that session.

**[0051]** This session number, Q value, handle, and slot counter data could be stored in arrays or matrices, in some embodiments.

**[0052]** In some embodiments, session number information, and last Q and last RN16 (Handle) information read from a tag for a given session is stored in a reader (or in the reader controller), and the reader is configured to selectively issue a command to resume a session. The command may be, for example, RESUME SESSION X, where X is a session number. The command passes session number, last Q and last Handle parameters from the reader to the tag from which those parameters were last read by the reader. The reader may then resume the inventory session that got interrupted.

**[0053]** In some embodiments, as is the case with the Gen 2 standard, the tags and readers have inventory rounds of  $2^Q$  slots. At the beginning of each round every selected tag sets one of their slot counters, associated with an inventorying, to a random number from 0 to  $2^Q-1$ .

**[0054]** As is the case with the Gen 2 standard, a user is able to include in an inventory round only tags that meet certain selection criteria. Appropriate combinations of Select

commands can be used to implement Boolean criteria within a tag population. This is why the term “selected” is used in the previous paragraph.

[0055] In operation, the reader sends a command that starts a slot. Any tag whose slot counter, for the present inventorying, is 0 sends a reply; all other tags decrease their slot counter by 1. This process is repeated in response to QueryRep commands. Q is the number of bits, or width of the slot counters.

[0056] In some embodiments, Q is, for example, an integer between 0 and 15, and the number of slots is between  $2^0$  and  $2^{15}$ . In other embodiments, Q values indicating a wider width of the slot counters can be used

[0057] As is the case in the Gen 2 specification, the tags support multiple sessions. Readers can use different sessions to independently inventory a common tag population. Tags associate a separate and independent “inventoried” flag to each of several readers. After singulating a tag, a reader may issue a command that causes the tag to invert its inventoried flag for that session. Unlike in the Gen 2 specification, tags may be inventoried by more than four readers or more than four sessions. A reader controller 82 may pass inventory session data from a first reader to a second reader, enabling a second reader to resume any session begun by a first reader. Such second reader may be multiplexed to a first reader’s antenna(s) under control of reader controller 82.

[0058] While the above discussion has been in terms of an improvement to the Gen 2 system and process, it will be readily apparent that the same concepts can be used to improve the Snodgrass and Wood, Jr. processes. For example, the Gen 2 handle or RN16 is substantially similar to the Random Number of the Snodgrass and Wood, Jr. patents. The Gen 2 Q parameter is substantially similar to the width of the Arbitration Mask of the Snodgrass and Wood, Jr. patents. Thus, in some embodiments, the system and process of any of the Snodgrass and Wood, Jr. patents is modified by adding storing a plurality of separate Arbitration Mask and Random Number values for multiple simultaneous arbitrations.

[0059] Thus, a system and method has been provided to increase the efficiency and effectiveness of each individual arbitration process and overall arbitration processes for

inventorying multiple concurrent tag populations, while preventing the case of infinite loop non-convergence of the processes.

**[0060]** In compliance with the statute, the subject matter disclosed herein has been described in language more or less specific as to structural and methodical features. It is to be understood, however, that the claims are not limited to the specific features shown and described, since the means herein disclosed comprise example embodiments. The claims are thus to be afforded full scope as literally worded, and to be appropriately interpreted in accordance with the doctrine of equivalents.



## CLAIMS

I/we claim,

1. An RFID (radio frequency identification device) tag comprising:  
a processor;  
a memory configured to store an identification number that distinguishes the tag from other tags;

a transponder coupled to the memory and the processor, the tag being configured to arbitrate by selecting a random number, in response to an inventory query from a reader, and to respond to the reader depending on the random number selected, the tag being further configured for multiple concurrent inventory session arbitrations with multiple readers by separately storing random numbers for respective inventory session arbitrations.

2. An RFID tag in accordance with claim 1 wherein the random numbers are stored in slot counters in the memory, wherein the tag includes separate slot counters for respective inventory session arbitrations, and wherein the tag is configured to reply to a reader if the value in a slot counter associated with an arbitration is zero.

3. An RFID tag in accordance with claim 2 wherein the slot counter is decremented in response to a command from a reader if the tag is in an arbitration with the reader.

4. An RFID tag in accordance with claim 1 and configured to store a plurality of Q values in the memory, where Q represents the size of a slot counter, wherein respective Q values are stored for respective inventory session arbitrations.

5. An RFID tag in accordance with claim 1 and configured to store a plurality of session values in the memory, identifying which reader is attempting to

inventory the tag, wherein respective session values are stored for respective inventory session arbitrations.

6. An RFID tag in accordance with claim 1 and configured to store a plurality of RN values in the memory, which are random numbers used by respective readers as handles, wherein respective RN values are stored for respective inventory session arbitrations.

7. An RFID tag in accordance with claim 1 wherein the transponder is a backscatter transponder.

8. An RFID reader configured to use arbitration to singulate unknown radio frequency identification tags in a field, different tags having different electronic product codes, the RFID reader comprising:

a transponder;

a processor; and

a memory configured to store, for respective inventory session arbitrations, handles for previously singulated tags, the handle being a nickname for a tag and being different from the tag's electronic product code, wherein, if an arbitration is interrupted, the reader is able to resume the arbitration without a need to cause singulated tags to generate new handles, even if another reader performs another arbitration session and causes tags to generate handles.

9. An RFID reader in accordance with claim 8 wherein the memory is configured to store last used Q values for respective inventory session arbitrations.

10. An RFID reader in accordance with claim 8 wherein the memory is configured to store last used arbitration mask values for respective inventory session arbitrations.

11. An RFID reader in accordance with claim 8 wherein the memory is configured to store slot counter values for previously singulated tags, wherein slot counter values are used to affect the order in which tags attempt to communicate with a reader in an arbitration.

12. An RFID reader in accordance with claim 9 wherein the memory is configured to store slot counter values for previously singulated tags, wherein slot counter values are used to affect the order in which tags attempt to communicate with a reader in an arbitration.

13. An RFID reader in accordance with claim 12 wherein the memory is configured to store a session number identifying an interrupted arbitration, and to associate stored Q values, handles, and slot counter values with the session number.

14. An RFID reader in accordance with claim 8 wherein the memory is configured to store session numbers identifying interrupted inventory session arbitrations, and to associate handles with the session numbers.

15. A system comprising:  
a plurality of RFID tags, respective RFID tags including:  
a processor;  
a memory configured to store an identification number that distinguishes the tag from other tags; and  
a transponder coupled to the memory and the processor, the tag being configured to arbitrate by selecting a random number, in response to an inventory query from a reader, and to respond to the reader depending on the random number selected, the tag being further configured for multiple concurrent inventory session arbitrations with multiple readers by separately storing random numbers for respective inventory session arbitrations; and  
a plurality of RFID readers configured to use arbitration to singulate radio frequency identification tags, respective RFID readers including:  
a transponder;  
a processor; and  
a memory configured to store, for respective arbitration sessions, handles for singulated tags, the handles being nicknames for tags and being different from a tag's identification number, wherein, if an arbitration is interrupted for one of the readers, that reader is able to resume and continue the arbitration without a need to cause singulated tags to generate new handles, even if another of the readers performs another arbitration and causes tags to generate handles.

16. A system in accordance with claim 15 wherein the random numbers are stored in slot counters, wherein the tags respectively include multiple slot counters, and wherein the tags are configured to reply to a reader if the value in a slot counter becomes zero during an arbitration.

17. A system in accordance with claim 16 wherein the slot counters are decremented in response to a predetermined command from one of the readers for tags in an arbitration session with that reader.

18. A system in accordance with claim 15 wherein respective RFID tags are configured to store a plurality of Q values, where Q represents the size of a slot counter, wherein respective Q values are stored for respective inventory session arbitrations.

19. A system in accordance with claim 15 wherein respective RFID tags are configured to store a plurality of session values, identifying which reader is attempting to inventory the tag wherein respective session values are stored for respective inventory session arbitrations.

20. A system in accordance with claim 15 wherein respective RFID tags are configured to store a plurality of RN values, wherein respective RN values are stored for respective inventory session arbitrations.

21. A system in accordance with claim 15 wherein the transponder of respective tags is a backscatter transponder.

22. A system in accordance with claim 15 wherein the respective readers are configured to store last used Q values for respective inventory session arbitrations.

23. A system in accordance with claim 18 wherein the respective readers are configured to store last used Q values for respective inventory session arbitrations.

24. A system in accordance with claim 15 wherein the respective readers are configured to store last used arbitration mask values for respective inventory session arbitrations.

25. A system in accordance with claim 15 wherein the respective readers are configured to store slot counter values for previously singulated tags, wherein slot counter values are used to affect the order in which tags attempt to communicate with a reader in an arbitration.

26. A system in accordance with claim 16 wherein the respective readers are configured to store slot counter values for previously singulated tags, wherein slot counter values are used to affect the order in which tags attempt to communicate with a reader in an arbitration.

27. A system in accordance with claim 23 wherein respective readers are configured to store slot counter values for previously singulated tags, wherein slot counter values are used to affect the order in which tags attempt to communicate with a reader in an arbitration.

28. A system in accordance with claim 15 wherein respective readers are configured to store a session number identifying an interrupted arbitration, and to associate stored Q values, handles, and slot counter values with the session number.

29. A system in accordance with claim 15 wherein respective readers are configured to store session numbers identifying interrupted inventory session arbitrations, and to associate handles with the session numbers.

30. A system in accordance with claim 15 and further comprising a reader controller coupled to at least some of the readers and configured to control those readers.

31. A system comprising:  
a plurality of RFID tags, respective RFID tags including:  
a processor;  
a memory configured to store an identification number that distinguishes the tag from other tags; and  
a transponder coupled to the memory and the processor, the tags being configured to arbitrate by selecting a random number, in response to an inventory query from a reader, and to respond to the reader depending on the random number selected, the tags being further configured to separately storing random numbers for respective inventory session arbitrations; and  
a plurality of RFID readers configured to use arbitration to singulate radio frequency identification tags, respective RFID readers including:  
a transponder;  
a processor; and  
a memory configured to store, for respective arbitration sessions, handles for singulated tags, the handles being nicknames for tags and being different from a tag's identification number, wherein, if an arbitration is interrupted for one of the readers, that reader is able to resume and continue the arbitration without a need to cause singulated tags to generate new handles, even if another of the readers performs another arbitration and causes tags to generate handles, wherein a second one of the readers is configured to selectively complete an arbitration started by a first one of the readers in response to having information from the inventory started by the first reader passed to the second reader.

32. A system in accordance with claim 31 and further comprising a reader controller coupled to at least some of the readers and configured to pass the information from the inventory started by the first reader to the second reader.

33. A system in accordance with claim 31 wherein the random numbers are stored in slot counters, wherein the tags respectively include multiple slot counters, and wherein the tags are configured to reply to a reader if the value in a slot counter becomes zero during an arbitration.

34. A system in accordance with claim 33 wherein respective RFID tags are configured to store a plurality of Q values, where Q represents the size of a slot counter, wherein respective Q values are stored for respective inventory session arbitrations.

35. A system in accordance with claim 34 wherein respective RFID tags are configured to store a plurality of session values, identifying which reader is attempting to inventory the tag wherein respective session values are stored for respective inventory session arbitrations.

36. A system in accordance with claim 35 wherein respective RFID tags are configured to store a plurality of RN values, wherein respective RN values are stored for respective inventory session arbitrations.

37. A system in accordance with claim 36 wherein the respective readers are configured to store last used Q values for respective inventory session arbitrations.

38. A system in accordance with claim 37 wherein the respective readers are configured to store last used arbitration mask values for respective inventory session arbitrations.



39. A system in accordance with claim 38 wherein the respective readers are configured to store slot counter values for previously singulated tags, wherein slot counter values are used to affect the order in which tags attempt to communicate with a reader in an arbitration.

40. A system in accordance with claim 39 wherein respective readers are configured to store a session number identifying an interrupted arbitration, and to associate stored Q values, handles, and slot counter values with the session number.

41. A method comprising:  
beginning a first RFID inventory session, using a first RFID reader, the first RFID reader being configured to store random numbers for tags that have been singulated;  
interrupting a first RFID inventory session;  
beginning a second RFID inventory session, and causing RFID tags to generate new random numbers for the second RFID inventory session, the RFID tags being configured to store the random numbers from the first inventory session during the second RFID inventory session; and  
resuming the first RFID inventory session.

42. A method in accordance with claim 41 wherein the first RFID reader is further configured to store arbitration mask information from the first inventory session, and to keep that information during the second inventory session.

43. A method in accordance with claim 42 wherein the first RFID reader is further configured to store session counter information for tags that were singulated during the first inventory session, and to keep that information during the second inventory session.

44. A method in accordance with claim 43 wherein RFID tags are configured to store information for multiple inventory session arbitrations.

45. A method comprising:  
providing a plurality of RFID tags, respective RFID tags including a processor; a memory configured to store an identification number that distinguishes the tag from other tags; and a transponder coupled to the memory and the processor, the tag being configured to arbitrate by selecting a random number, in response to an inventory query from a reader, and to respond to the reader depending on the random number selected, the tags being further configured to separately store random numbers for respective inventory session arbitrations;  
providing a plurality of RFID readers configured to use arbitration to singulate radio frequency identification tags, respective RFID readers including a transponder; a processor; and a memory configured to store, for respective arbitration sessions, handles for singulated tags, the handles being nicknames for tags and being different from a tag's identification number;  
interrupting an arbitration for a first one of the readers; and  
resuming and continuing the arbitration for the first one of the readers without a need to cause singulated tags to generate new handles.

46. A method in accordance with claim 45 and comprising storing the random numbers in slot counters, wherein the tags respectively include multiple slot counters, and respective tags replying to a reader if the value in the tag's slot counter becomes zero during an arbitration.

47. A method in accordance with claim 46 and comprising respectively decrementing the slot counters in response to a predetermined command from one of the readers for tags in an arbitration session with that reader.

48. A method in accordance with claim 45 and comprising storing in respective RFID tags a plurality of Q values, where Q represents the size of a slot counter, wherein respective Q values are stored for respective inventory session arbitrations.

49. A method in accordance with claim 45 and comprising storing, in respective RFID tags, a plurality of session values that identify which reader is attempting to inventory the tag, wherein respective session values are stored for respective inventory session arbitrations.

50. A method in accordance with claim 45 and comprising storing, in respective RFID tags, a plurality of RN values, wherein respective RN values are stored for respective inventory session arbitrations.

51. A method in accordance with claim 45 and comprising storing, in one of the readers, last used Q values for an arbitration.

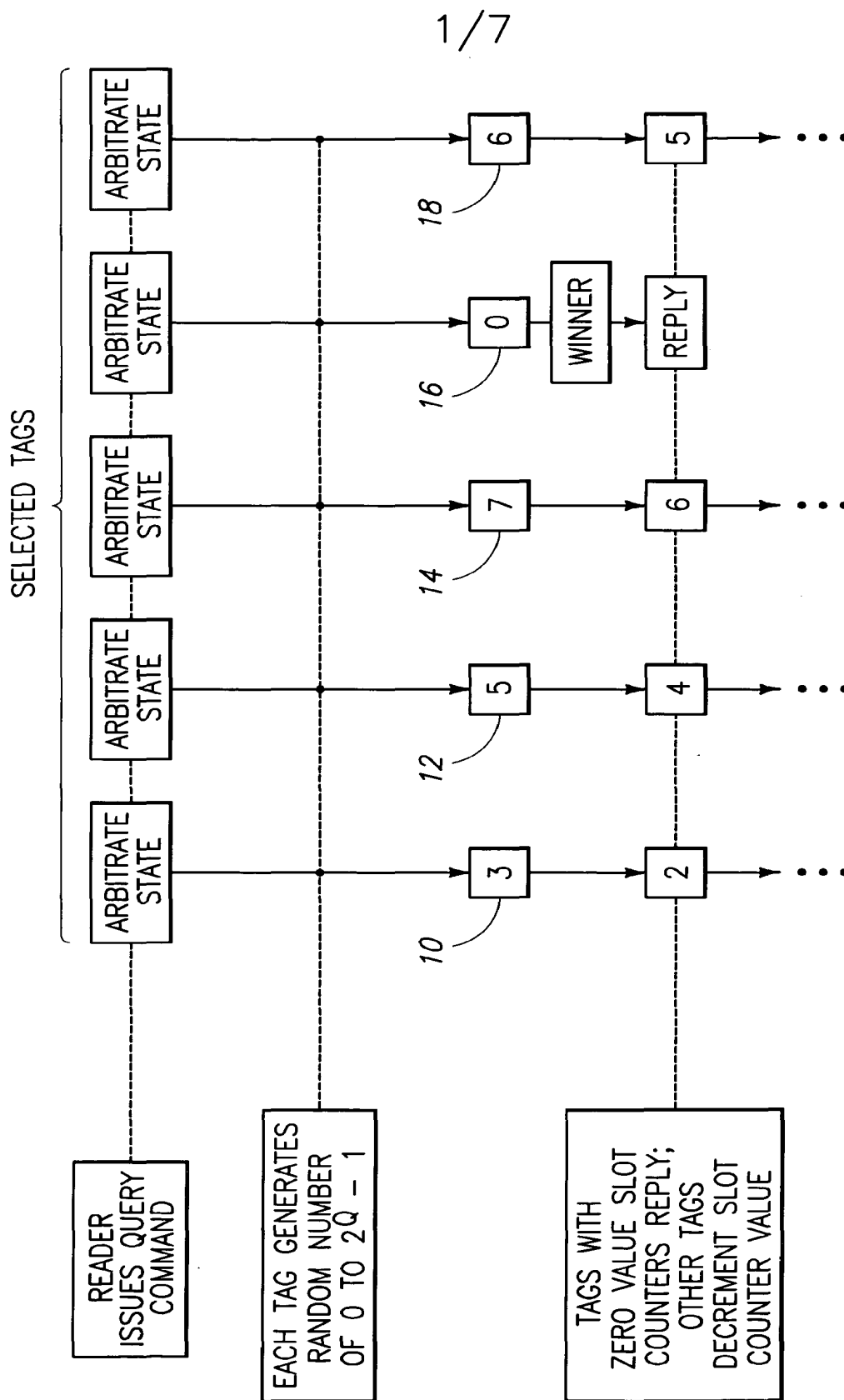
52. A method in accordance with claim 45 and comprising storing, in one of the readers, last used arbitration mask values for an arbitration.

53. A method in accordance with claim 45 and comprising storing, in one of the readers, slot counter values for previously singulated tags.

54. A method in accordance with claim 45 and comprising storing, in one of the readers, a session number identifying an interrupted arbitration, and to associate stored Q values, handles, and slot counter values with the session number.

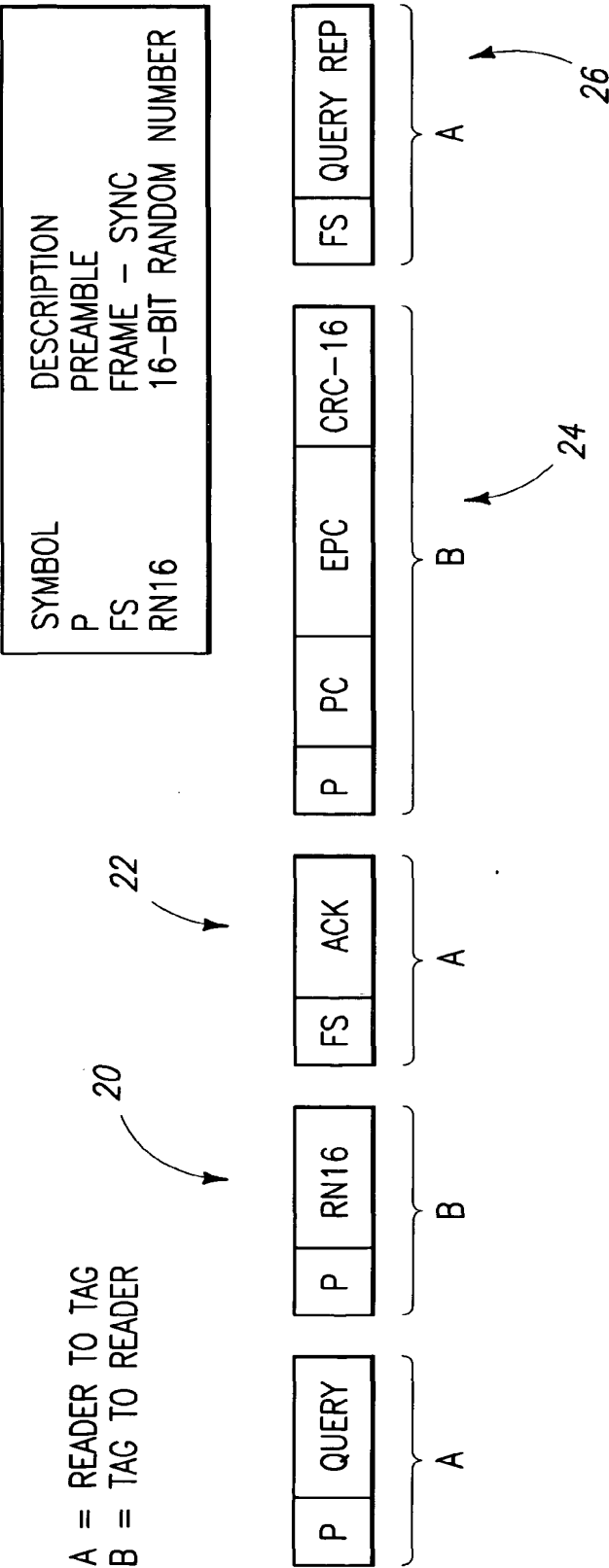
55. A method in accordance with claim 45 and further comprising coupling a reader controller to at least some of the readers controlling those readers using the reader controller.

56. An RFID reader comprising:  
memory configured to store a session number, and to store last Q, and last RN16 information received from a tag in an arbitration session; and  
a transponder coupled to the memory and configured to selectively send a command to the tag to resume an interrupted arbitration session, the command including parameters for the session number, the last Q, and the last RN16 read from the tag by the reader.



*FIG. 1*

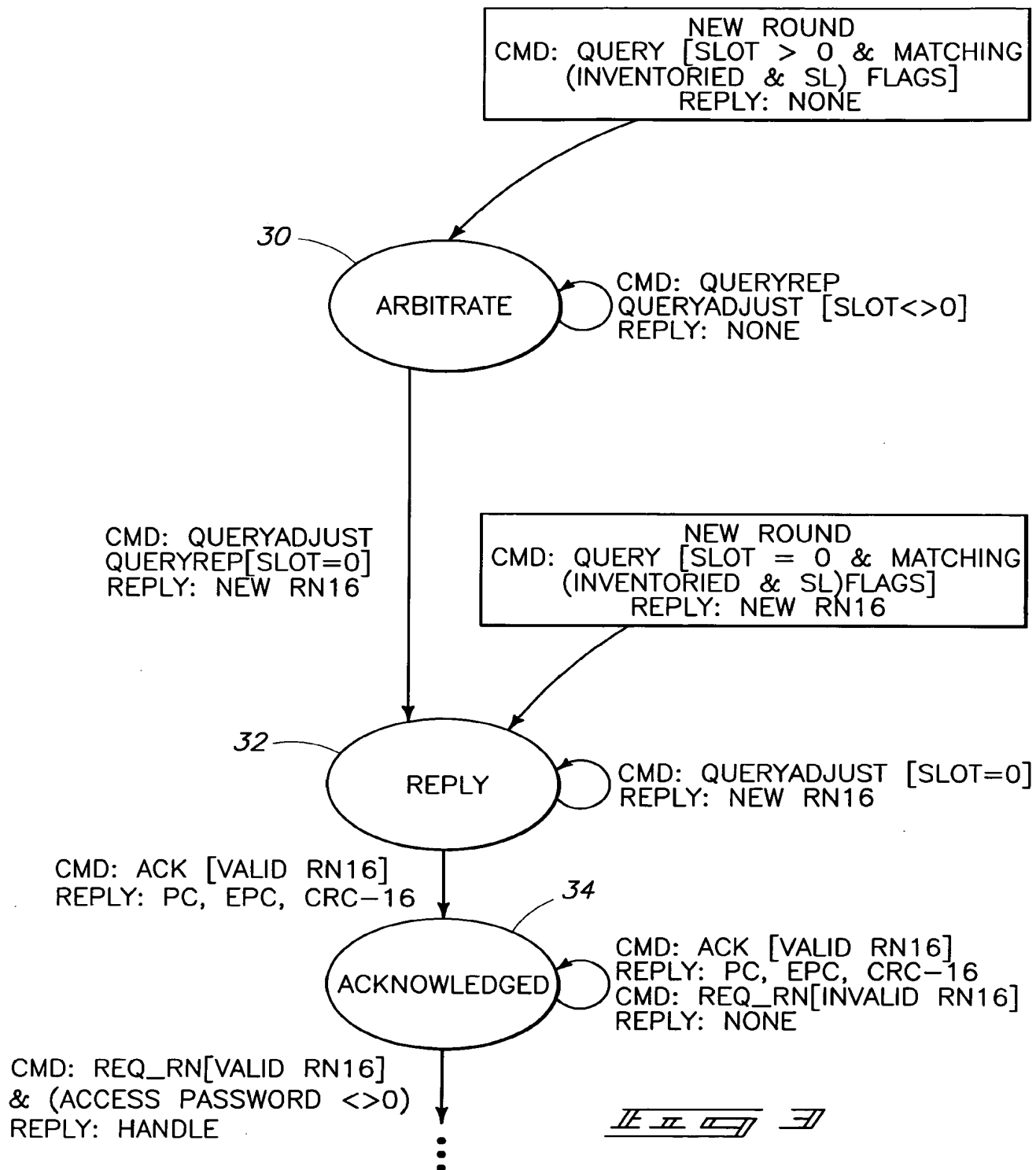
2/7



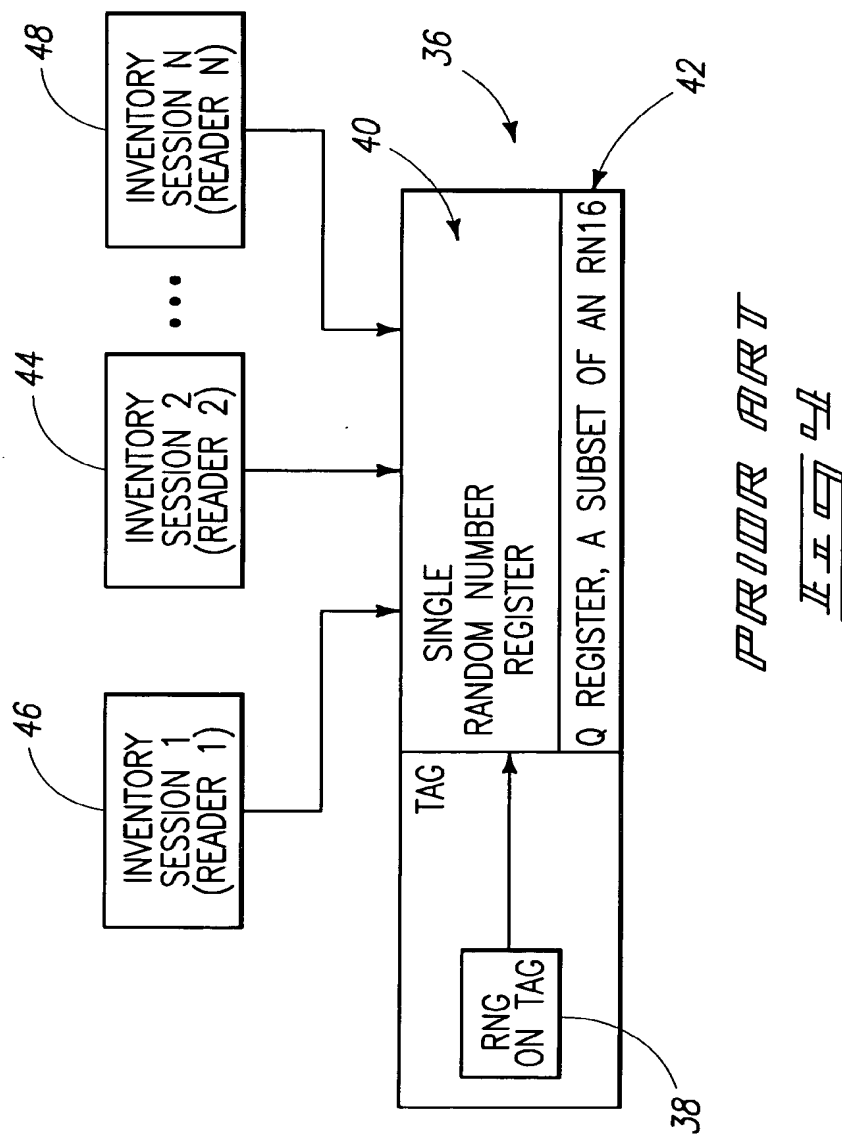
*PRIOR ART*

II

3/7



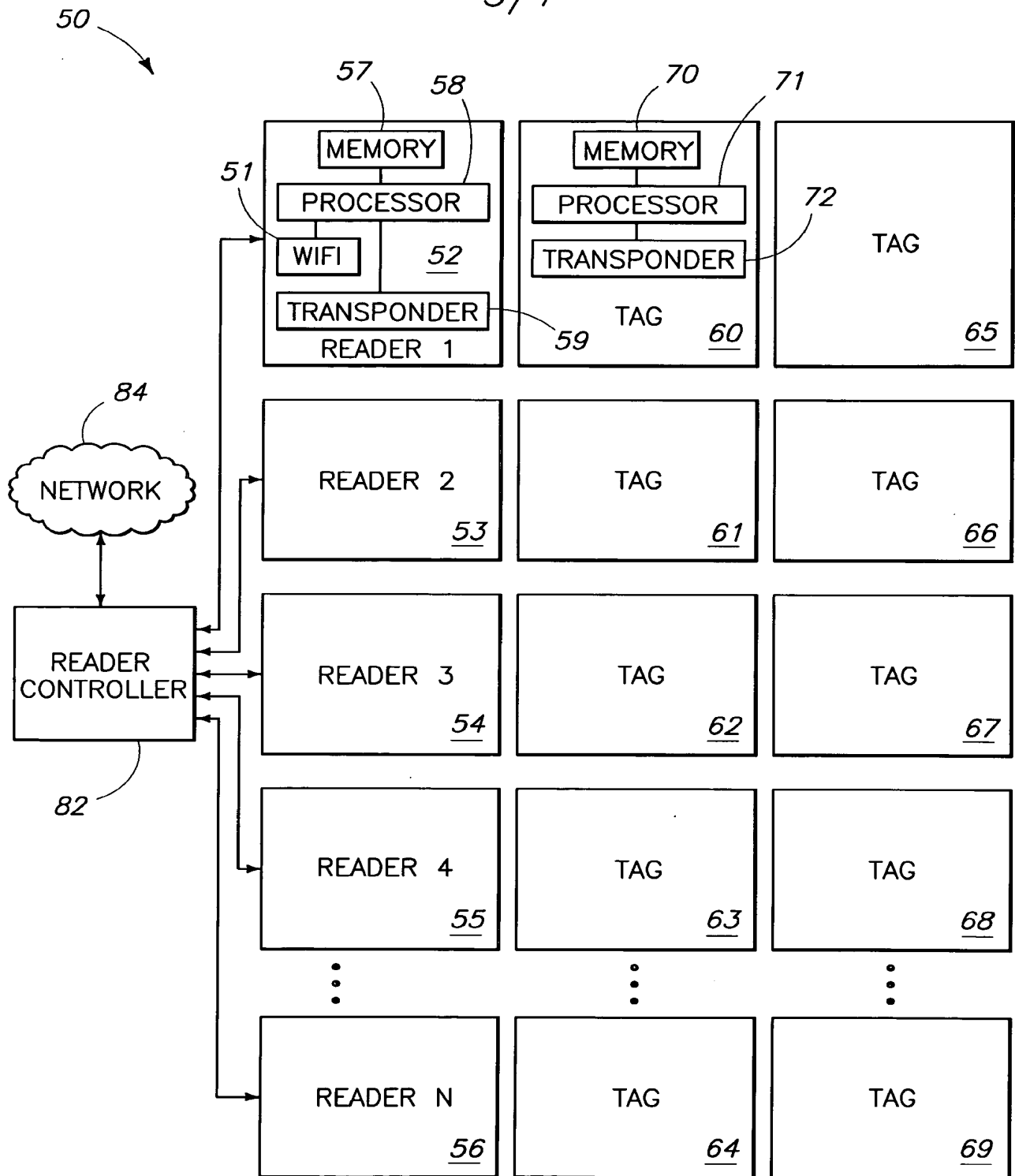
4/7



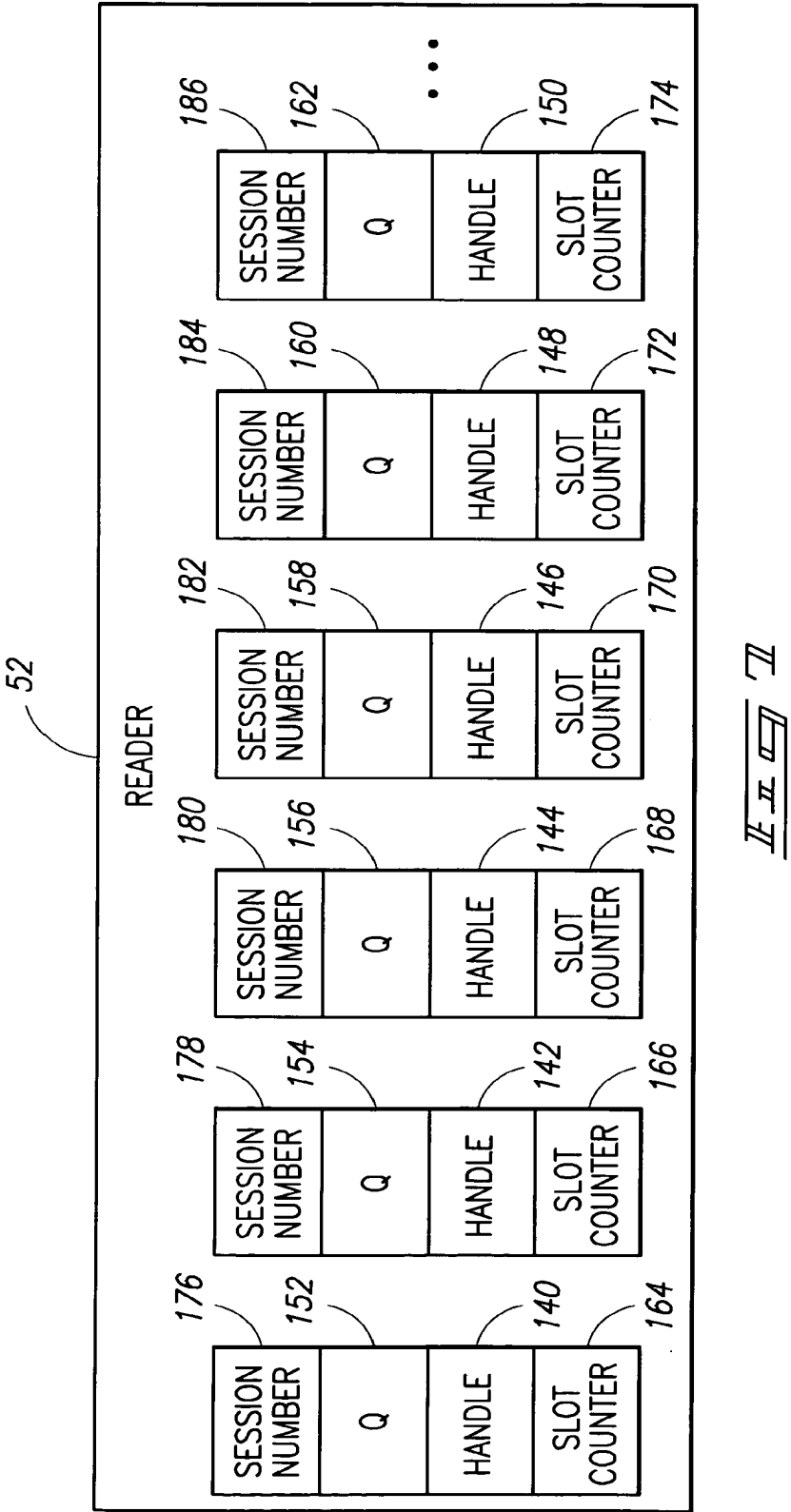
*PRIOR ART*



5/7







## INTERNATIONAL SEARCH REPORT

International application No.  
**PCT/US2008/050630****A. CLASSIFICATION OF SUBJECT MATTER****G06K 19/07(2006.01)i**

According to International Patent Classification (IPC) or to both national classification and IPC

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)

IPC 8 G06K, G01S, H04L, H04J, H04K

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean Utility models and applications for Utility models since 1975

Japanese Utility models and applications for Utility models since 1975

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKIPASS (KIPO internal) "arbitration", "singulation", "rfid", "tag", "reader"

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 5627544 A (SNODGRASS, C.K. et al.) 06 May 1997 See abstract, claims 1-53, figures 1-15.	56
A		1-55
A	US 5231646 A (HEATH, W.A. et al.) 27 July 1993 See abstract, claims 1-18, figures 1-13.	1-56
A	US 6275476 B1 (WOOD, JR., C.W.) 14 August 2001 See abstract, claims 1-3, figures 1-6.	1-56
A	KR 10-2005-0013227 A (MOTOROLA INC.) 03 February 2005 See abstract, claims 1-21, figures 1-5.	1-56

☐ Further documents are listed in the continuation of Box C.☒ See patent family annex.

\* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&amp;" document member of the same patent family

Date of the actual completion of the international search

27 JUNE 2008 (27.06.2008)

Date of mailing of the international search report

**27 JUNE 2008 (27.06.2008)**

Name and mailing address of the ISA/KR

Korean Intellectual Property Office  
Government Complex-Daejeon, 139 Seonsa-ro, Seo-  
gu, Daejeon 302-701, Republic of Korea

Facsimile No. 82-42-472-7140

Authorized officer

LEE, Seung Joo

Telephone No. 82-42-481-8186



**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2008/050630**

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5627544 A	06.05.1997	None	
US 5231646 A	27.07.1993	None	
US 6275476 B1	14.08.2001	AU 1999-32859 A1 EP 1072128 A1 US 2001-0014090 A1 US 2005-0207364 A1 US 2006-0056325 A1 US 2006-0209781 A1 WO 99-43127 A1	06.09.1999 31.01.2001 16.08.2001 22.09.2005 16.03.2006 21.09.2006 26.08.1999
KR 10-2005-0013227 A	03.02.2005	AU 2003-243195 A1 CN 1330140 C CN 1663187 A EP 1518365 A1 JP 2005-530454 A US 2003-0235184 A1 WO 2004-002071 A1	06.01.2004 01.08.2007 31.08.2005 30.03.2005 06.10.2005 25.12.2003 31.12.2003